



# Etude Comparative des Architectures des Microprocesseurs Intel Pentium et PowerPC 601

André Seznec, Thierry Vauléon

## ► To cite this version:

André Seznec, Thierry Vauléon. Etude Comparative des Architectures des Microprocesseurs Intel Pentium et PowerPC 601. [Rapport de recherche] RR-2320, INRIA. 1994. inria-00074354

**HAL Id: inria-00074354**

**<https://inria.hal.science/inria-00074354>**

Submitted on 24 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

***Étude Comparative des Architectures des  
Microprocesseurs Intel Pentium et PowerPC 601***

André Seznec, Thierry Vauléon

**N° 2320**

Août 1994

PROGRAMME 1



***apport  
de recherche***





## Étude Comparative des Architectures des Microprocesseurs Intel Pentium et PowerPC 601

André Seznec, Thierry Vauléon

Programme 1 — Architectures parallèles, bases de données, réseaux et systèmes distribués  
Projet CAPS

Rapport de recherche n° 2320 — Août 1994 — 108 pages

**Résumé :** L'évolution de l'architecture des microprocesseurs est très rapide ; dans ce rapport nous présentons de manière synthétique les architectures de deux microprocesseurs introduits en 1993 : l'Intel Pentium et le PowerPC 601 ; ce dernier a été conçu conjointement par Apple, IBM et Motorola. Nous avons regroupé de manière uniforme les informations disponibles sur ces deux microprocesseurs ; les convergences et les divergences d'approche des différents constructeurs sont ainsi dégagées. Les évolutions par rapport à la génération précédente sont aussi soulignées.

**Mots-clé :** veille technologique, microprocesseur, architecture, calcul flottant, hiérarchie mémoire, support des systèmes d'exploitation, interface bus

*(Abstract: pto)*

ce travail a été partiellement soutenu par la DRET par la convention DRET-INRIA n° 93/082

## **Intel Pentium and PowerPC 601 Architectures : a comparative study**

**Abstract:** The Intel Pentium and PowerPC 601, jointly made by Apple, IBM and Motorola, are recent microprocessors that have been introduced in 1993. In this report, we present in a uniform and synthetic form, the informations which are available on these two microprocessors. We try to point out the convergence of all designs on some features and some divergences on others features.

**Key-words:** technological watch, microprocessor, architecture, floating point unit, caches, memory management unit, system interface

# Table des matières

<b>1</b>	<b>Jeu d'instructions</b>	<b>11</b>
1.1	Introduction . . . . .	11
1.2	Types de données . . . . .	12
1.3	Modes d'adressage . . . . .	12
1.4	Formats des instructions . . . . .	14
1.4.1	Format des instructions du PowerPC 601 . . . . .	15
1.4.2	Format des instructions du Pentium . . . . .	15
1.5	Instructions . . . . .	18
1.5.1	Les instructions de transfert de données . . . . .	18
1.5.2	Instructions arithmétiques et logiques . . . . .	20
1.5.3	Les instructions de contrôle de flot . . . . .	21
1.5.4	Les opérations sur les chaînes de caractères . . . . .	22
1.5.5	Les instructions pour les langages structurés en blocs . . . . .	23
1.5.6	Les instructions flottantes . . . . .	23
1.5.7	Différences entre le jeu d'instructions du PowerPC 601 et la norme PowerPC . . . . .	24
1.6	Conclusion . . . . .	25
<b>2</b>	<b>Architecture : organisation générale</b>	<b>26</b>
2.1	Introduction . . . . .	26
2.2	Synoptique des unités fonctionnelles . . . . .	26
2.2.1	Architecture du PowerPC 601 . . . . .	26
2.2.2	Architecture du Pentium . . . . .	30
2.3	Pipeline . . . . .	33
2.3.1	Groupeement des instructions . . . . .	34
2.3.2	Alignement des instructions . . . . .	35
2.3.3	Pipeline du PowerPC 601 . . . . .	37
2.3.4	Pipeline du Pentium . . . . .	40

2.3.5	Traitement de ruptures de séquence . . . . .	41
2.4	Conclusion . . . . .	47
<b>3</b>	<b>Unité de calcul flottant</b>	<b>49</b>
3.1	Introduction . . . . .	49
3.2	Architecture . . . . .	50
3.2.1	Unité flottante du PowerPC 601 . . . . .	50
3.2.2	Unité flottante du Pentium . . . . .	50
3.3	Pipeline . . . . .	52
3.3.1	Pipeline flottant du PowerPC 601 . . . . .	52
3.3.2	Pipeline flottant du Pentium . . . . .	53
3.4	Exceptions flottantes . . . . .	55
3.4.1	Exceptions flottantes du PowerPC 601 . . . . .	55
3.4.2	Exceptions flottantes du Pentium . . . . .	55
3.5	Conclusion . . . . .	55
<b>4</b>	<b>Hierarchie mémoire</b>	<b>57</b>
4.1	Introduction . . . . .	57
4.2	Caches internes . . . . .	57
4.2.1	Accès aux caches . . . . .	58
4.2.2	Placement des données . . . . .	60
4.2.3	Tailles et formats des lignes . . . . .	62
4.2.4	Stratégies de recopie en mémoire . . . . .	62
4.2.5	Les tampons d'écritures . . . . .	63
4.2.6	Adressages des caches internes . . . . .	65
4.3	Second niveau de cache . . . . .	65
4.4	Conclusion . . . . .	67
<b>5</b>	<b>Support des systèmes d'exploitation</b>	<b>68</b>
5.1	Introduction . . . . .	68
5.2	Espace virtuel . . . . .	68
5.2.1	Caractéristiques . . . . .	69
5.2.2	La segmentation . . . . .	70
5.3	Traduction d'adresse . . . . .	75
5.3.1	Mécanisme de traduction d'adresse du PowerPC 601 . . . . .	75
5.3.2	Mécanisme de traduction d'adresse du Pentium . . . . .	80
5.3.3	Caches de traduction d'adresses . . . . .	81
5.4	Sécurité . . . . .	82

5.4.1	Sécurité sur le PowerPC 601 . . . . .	82
5.4.2	Sécurité sur le Pentium . . . . .	82
5.5	Support au mode de fonctionnement multi-tâches . . . . .	83
5.6	Conclusion . . . . .	83
<b>6</b>	<b>Interface bus</b>	<b>85</b>
6.1	Introduction . . . . .	85
6.2	Protocole de transfert de base . . . . .	86
6.2.1	Transfert simple . . . . .	86
6.2.2	Transfert groupé . . . . .	86
6.2.3	Protocole d'acquisition du bus . . . . .	87
6.2.4	Pipeline des transferts . . . . .	89
6.3	Transfert de cohérences des caches . . . . .	93
6.4	Conclusion . . . . .	93
<b>7</b>	<b>Support multiprocesseur</b>	<b>95</b>
7.1	Introduction . . . . .	95
7.2	Cohérence de caches . . . . .	95
7.3	Support de synchronisation . . . . .	96
7.3.1	Accès à la mémoire partagée . . . . .	98
7.3.2	Ordre des lectures/écritures . . . . .	99
7.4	Support aux systèmes à redondance (ou maître-esclave) . . . . .	100
7.5	Conclusion . . . . .	100



# Table des figures

1.1	Principaux formats des instructions du PowerPC 601 . . . . .	16
1.2	Format des instructions du Pentium . . . . .	17
2.1	Architecture du PowerPC 601 . . . . .	27
2.2	File d'instructions du PowerPC 601 . . . . .	29
2.3	Architecture de l'Intel Pentium . . . . .	31
2.4	schéma fonctionnel des pipelines entiers de l'Intel Pentium . . . . .	36
2.5	pire cas de chargement d'une instruction . . . . .	37
2.6	Pipeline du PowerPC 601 . . . . .	38
2.7	Machine d'états de l'algorithme de prédiction de branchement du Pentium . . . . .	46
3.1	Architecture de l'unité flottante du Pentium . . . . .	51
4.1	Description fonctionnelle d'une mémoire entrelacée à deux ports . . . . .	59
4.2	Organisation du cache du PowerPC 601 . . . . .	61
4.3	Unité mémoire . . . . .	63
5.1	Modèle de mémoire uniforme du Pentium . . . . .	72
5.2	Modèle de mémoire uniforme protégée du Pentium . . . . .	72
5.3	Modèle de mémoire segmentée du Pentium . . . . .	73
5.4	schéma explicatif des mécanismes de traduction d'adresse . . . . .	74
5.5	Les différents mécanismes de traduction d'adresse sur le PowerPC 601 . . . . .	76
5.6	Traduction de l'adresse d'une page dans le PowerPC 601 . . . . .	77
5.7	Description détaillée du mécanisme de pagination du PowerPC 601 . . . . .	79
5.8	Mécanisme de traduction d'adresse du Pentium . . . . .	80
6.1	Protocole de transfert de lectures groupées avec le Pentium . . . . .	88
6.2	système utilisant un maître central . . . . .	89
6.3	Lectures groupées en mode pipeline par deux PowerPC 601 . . . . .	90

6.4	Lectures groupées en mode pipeline avec le Pentium . . . . .	91
6.5	Lectures groupées en mode pipeline avec le PowerPC 601 . . . . .	92
7.1	Protocole MESI à invalidation sur écriture . . . . .	97
7.2	Protocole SI à invalidation sur écriture . . . . .	98

# Liste des tableaux

1.1	Exemple d'utilisation de la préincrémentation . . . . .	13
3.1	latences et délais entre les instructions flottantes en format double précision . . .	53
7.1	vue d'ensemble de différents microprocesseurs du marché . . . . .	104
7.2	Largeur des chemins de données . . . . .	105
7.3	Caractéristiques des caches internes . . . . .	105
7.4	Mémoire virtuelle/Mémoire physique . . . . .	105
7.5	Pipeline . . . . .	106
7.6	Pipeline flottant . . . . .	106

# Introduction

Dans les deux rapports précédents [1, 2], nous présentions les microprocesseurs Mips R3000, Sun Sparc et IBM RS6000 [1] et les microprocesseurs Mips R4000, T.I. Supersparc et Dec 21064 [2]. Dans ce rapport, nous nous intéressons aux microprocesseurs Intel Pentium et PowerPC601. Les microprocesseurs étudiés précédemment dans [1] et [2] avaient tous une architecture RISC (*Reduced Instruction Set Computer*) [3]. Nous étudions ici un microprocesseur RISC, le PowerPC 601 et un microprocesseur CISC (*Complex Instruction Set Computer*), le Pentium.

## Contexte

Le but avoué du consortium PowerPC (Apple, Motorola et IBM) est de briser le quasi-monopole des processeurs de la famille xx86 sur le marché des PCs. Des facteurs externes à l'architecture et à la technologie (compatibilité, puissance commerciale, stratégie des fabricants de PC, stratégie des éditeurs de logiciel, ...) décideront sans doute de l'issue de ce défi. L'étude que nous présentons ici se limite aux points techniques des architectures (jeu d'instructions, implémentations, interface externe etc).

## Un peu d'histoire

Depuis l'introduction du microprocesseur 8086 en 1978, l'architecture xx86 a évolué à travers plusieurs améliorations tant au niveau architectural que technologique. Ainsi le i486, introduit en 1989, intègre sur une seule puce, en plus de toutes les fonctionnalités d'un processeur entier, une unité flottante et un cache. Le Pentium, quant à lui, intègre deux unités entières, une unité flottante totalement pipelinée ainsi que deux caches séparés.

*Tout au long de ces 16 années, une compatibilité binaire ascendante a été maintenue.*

Le PowerPC 601, premier microprocesseur de la famille PowerPC, a été développé par IBM, Motorola et Apple. Son architecture a été fortement inspirée de celle de l'IBM POWER (1990) qui a été simplifiée pour enlever quelques instructions superflues. Ce processeur n'a donc pas une histoire aussi pesante que l'Intel Pentium, ce qui a grandement simplifié sa conception.

## **Présentation générale**

Dans ce rapport, nous présentons de manière uniforme les deux processeurs étudiés en tentant de souligner et si possible d'expliquer les divergences d'approche sur de nombreux points.

Les informations présentées ont été collectées dans la documentation technique des deux processeurs [5, 6, 7],[4].

Aucune mesure de performance n'a été conduite, ces performances dépendant de nombreux facteurs matériels et logiciels extérieurs à cette étude. Des mesures de performances (SPEC-marks), présentées dans le tableau 7.1, sont données à titre purement indicatif et doivent être interprétées avec beaucoup de prudence (type de hiérarchie mémoire, optimisation des fautes par le compilateur, système d'exploitation employé, etc...).

# Chapitre 1

## Jeu d'instructions

### 1.1 Introduction

Les deux microprocesseurs étudiés ont des jeux d'instructions radicalement différents. Ils doivent cependant tous les deux satisfaire à une compatibilité binaire avec leurs prédécesseurs qui représentaient des concepts opposés; du fait de son histoire beaucoup plus longue, cette particularité est plus pesante pour le Pentium que pour le PowerPC 601.

En effet, le succès des architectures xx86 oblige Intel à préserver une totale compabilité ascendante entre ses différents microprocesseurs voués au marché des PC (*Personal Computer*). De ce fait, le jeu d'instructions du Pentium contient toutes les instructions du i486 auxquelles ont été ajoutées quelques instructions système et utilisateur.

Le PowerPC 601 est le premier microprocesseur de la famille PowerPC, mais son jeu d'instructions est fortement inspiré de celui du microprocesseur POWER. Il possède de ce fait, quelques instructions présentes dans le jeu d'instructions de l'IBM POWER mais absentes dans la norme PowerPC. Le PowerPC 601, comme le Pentium, possède un jeu d'instructions 32 bits (c'est-à-dire que les adresses sont calculées sur 32 bits). Dans la famille PowerPC, seul le 620 possèdera un jeu d'instructions 64 bits.

Le jeu d'instructions du PowerPC 601 est à trois opérandes (quatre pour les instructions de multiplication-addition) : le résultat d'une opération est rangé dans un registre dont le numéro est explicitement codé dans l'instruction. Sur le Pentium, toute opération est destructrice : le résultat est rangé dans l'une des sources de l'instruction (registre ou emplacement mémoire).

Dans ce chapitre, nous étudions les deux jeux d'instructions en insistant particulièrement sur les différences entre les microprocesseurs d'architecture RISC (Reduced Instruction Set Computer) et ceux d'architecture CISC (Complex Instruction Set Computer). Au niveau des instructions, ces différences se situent sur leur nombre, leurs formats et les types d'instructions

implémentées. À noter que le PowerPC 601 implémente un grand nombre d'instructions absentes sur la plupart des autres microprocesseurs d'architecture RISC.

## 1.2 Types de données

Les deux microprocesseurs étudiés permettent d'accéder en mémoire des opérandes de différents formats :

- pour les entiers : octets, demi-mots (16 bits) et mots (32 bits)<sup>1</sup>
- pour les flottants : mots et double-mots

Trois types de données sont utilisés par le Pentium :

- les entiers binaires : trois types composent ce groupe, ils ne varient que par la taille des données (8, 16 et 32 bits). Les nombres signés sont représentés en complément à deux.
- les entiers décimaux : ils sont chargés sous la forme de paquets d'octets. Les nombres négatifs sont représentés en complément à un.
- les nombres réels (flottants)

## 1.3 Modes d'adressage

Les modes d'adressage courants peuvent être réalisés en une seule instruction avec les deux microprocesseurs étudiés :

- l'adressage absolu : immédiat
- l'adressage indirect : registre + 0
- l'adressage basé : registre + immédiat
- l'adressage indexé : registre + registre

En plus, le Pentium et le PowerPC 601 possèdent chacun un mode d'adressage supplémentaire : l'adressage avec préincrément pour le PowerPC 601 et l'adressage basé indexé pour le Pentium.

---

1. Nous nous détachons volontairement des terminologies employées par les constructeurs

**Adressage avec préincrémentation du PowerPC 601 :** le PowerPC 601 met en œuvre la technique de la préincrémentation. Cette technique était déjà utilisée dans le microprocesseur POWER : elle permet de mettre à jour le registre de base par le résultat du calcul d'adresse au cours de l'instruction.

Cet adressage permet, dans certains cas, de gagner une instruction à chaque accès. Le tableau 1.3 illustre la différence entre un microprocesseur utilisant la préincrémentation et un autre ne l'utilisant pas dans le cas de l'accès à l'élément d'un tableau au sein d'une boucle. Cette option est particulièrement importante en calcul numérique car elle permet d'obtenir des performances très proches d'un accès mémoire par cycle avec une seule unité entière.

Sans préincrémentation:

```
R1 ← R1+4      ; Remise à jour du registre  
                ; permettant l'adressage.  
LD R1, R2      ; l'élément suivant est accédé  
                ; et placé dans R2.
```

Avec préincrémentation:

```
LDU R1+4, R2   ; La mise à jour de R1 est effectuée
```

TAB. 1.1 - *Exemple d'utilisation de la préincrémentation*

Le chargement d'une donnée qui utilise la préincrémentation produit deux résultats : la donnée lue et le registre d'adresse mis à jour (pour le prochain accès). Pour pouvoir être mise en œuvre, la technique de préincrémentation pour les lectures d'entiers nécessite donc la présence d'un port d'écriture supplémentaire sur le banc de registres entiers. L'implémentation de cette technique uniquement sur les instructions de lecture/écriture flottantes et les instructions d'écriture entières aurait, par contre été immédiate puisque, lors de telles opérations, le banc de registre entier n'est pas accédé en écriture<sup>2</sup>.

**Adressage basé indexé du Pentium :** sur le Pentium, le mode de calcul de l'adresse est déterminé dans l'octet suivant immédiatement le code opération (*OPCODE*). En plus du segment adressé qui peut être défini de façon explicite par le préfixe de l'instruction ou bien de façon

---

2. Dans notre enthousiasme pour cette remarque simple, nous avons indiqué par erreur dans [2] que les architectures du POWER avaient utilisé cette astuce



implicite en utilisant, par défaut, le contenu du registre de segment adéquat, quatre éléments servent au calcul de l'adresse :

- un déplacement
- un registre de base
- un registre d'index
- un facteur d'échelle qui permet de multiplier par 2, 4 ou 8 le contenu du registre d'index.

La taille des instructions n'étant pas fixe, toutes les composantes utiles au calcul de l'adresse peuvent être présentes dans la même instruction, ce qui a pour effet d'obtenir l'adressage basé indexé :  $\text{base} + (\text{index} * \text{facteur d'échelle}) + \text{constante}$ . Intel annonce que ce mode d'adressage est particulièrement efficace pour adresser des tableaux à deux dimensions d'éléments de 2, 4 ou 8 octets. Il permet, à priori, au programmeur d'adresser de façon explicite un tableau à deux dimensions (ex :  $A[7][i]$ ). Cependant, comme la constante (le déplacement) doit être connue à la génération de code<sup>3</sup>, l'utilité de ce mode d'adressage semble limitée.

## 1.4 Formats des instructions

Le jeu d'instructions du **PowerPC 601** est de type RISC. Il est conçu pour simplifier au maximum le décodage et le contrôle des instructions en utilisant un format d'instruction simple; les instructions sont de taille fixe, ce qui simplifie leur rangement en mémoire.

Le jeu d'instructions du **Pentium** est de type CISC. De telles architectures tentent d'implémenter le plus possible de primitives employées par les langages de haut niveau (après traduction binaire). Les codes assembleurs, c'est-à-dire ceux compris par la machine, ressemblent donc aux codes source initiaux écrits par les programmeurs. De tels microprocesseurs possèdent de ce fait des instructions très variées et complexes. Ainsi, en particulier, les instructions du Pentium n'ont pas toutes la même taille.

Les formats des instructions des deux microprocesseurs étudiés diffèrent aussi par le nombre d'opérandes qu'ils permettent d'adresser. Chaque instruction du Pentium peut adresser deux opérandes, un registre et un emplacement mémoire (ou un autre registre); le résultat de l'opération est stocké dans un de ces deux emplacements.

Par contre, la plupart des instructions arithmétiques du PowerPC 601 adressent trois opérandes, toutes dans des registres. Les instructions de multiplication-addition flottantes adressent quatre opérandes ( $A \times B + C \rightarrow D$ ).

---

3. sauf pour les adeptes du *Self modifying code*

#### 1.4.1 Format des instructions du PowerPC 601

Toutes les instructions du PowerPC 601 ont une taille de 32 bits; de ce fait, elles sont alignées en mémoire sur les frontières de mots.

La figure 1.1 montre les principaux formats des instructions du PowerPC 601.

Contrairement à d'autres microprocesseurs d'architecture plus proches des fondements RISC (ex : Mips R4000, Dec Alpha...), le PowerPC 601, comme l'IBM POWER, possède un grand nombre de formats d'instructions (plus de 10). Cette caractéristique a pour principale conséquence de compliquer l'unité de décodage du microprocesseur.

On peut noter que les instructions de lecture/écriture (*load/store*) possèdent les mêmes formats que les instructions arithmétiques.

#### 1.4.2 Format des instructions du Pentium

La figure 1.2 montre l'occurrence des différents champs composant les instructions du Pentium. Il faut noter que seul le code opération est obligatoirement présent. Six champs principaux servent à composer une instruction qui est d'une taille inférieure ou égal à 15 octets. La plupart des instructions n'accédant pas d'opérandes, ou accédant toujours le même registre, tient sur un octet (ex : *DX < -* extension signée de *AX*, *HALT*; *FSQRT*: racine carrée de sommet de pile flottante...) tandis qu'une instruction adressant une opérande en mémoire, utilisant tous les préfixes et le mode d'adressage basé indexé peut tenir sur 15 octets. Les six champs d'une instruction du Pentium sont décrits dans l'ordre de leur occurrence dans la liste suivante :

- **les préfixes** : leur fonction est de modifier l'opération réalisée par l'instruction. Ils peuvent être jusqu'à quatre dans une même instruction :
  - *le préfixe de l'instruction* : les instructions manipulant des chaînes de caractères (*String Instructions*) peuvent contenir un préfixe de type *Repeat* qui a pour conséquence d'exécuter l'instruction tant qu'une condition n'est pas observée (ex : répéter jusqu'à opérande=0). Le préfixe d'instruction *LOCK* peut, quant à lui, être codé dans toutes les instructions accédant la mémoire, y compris les instructions arithmétiques; il permet d'effectuer des accès à la mémoire de façon atomique et, ainsi, d'implémenter des protocoles d'exclusion mutuelle.
  - *le préfixe de la taille de l'adresse* : ce préfixe est utilisé pour définir le mode de fonctionnement du Pentium. En effet, le Pentium peut aussi exécuter des codes prévus pour le 8086 (architecture 16 bits). Deux tailles d'adresse sont donc possibles, 16 bits et 32 bits. Chacune peut être choisie par défaut. La présence du préfixe dans une instruction permet d'utiliser l'autre taille.